**PENGEMBANGAN MODUL KEUANGAN APLIKASI PEMBUKUAN AKTA NOTARIS DAN PPAT**

**MENGGUNAKAN YII FRAMEWORK**

**PROPOSAL PROYEK AKHIR**



**Oleh :**

**ANDY FERBIANTORO**

**NIM. 361755401010**

**PROGRAM STUDI DIPLOMA III**

**TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANYUWANGI**

**2019**

**PENGEMBANGAN MODUL KEUANGAN APLIKASI PEMBUKUAN AKTA NOTARIS DAN PPAT**

**MENGGUNAKAN YII FRAMEWORK**

# HALAMAN JUDUL

**PROPOSAL PROYEK AKHIR**



Proyek Akhir Ini Dibuat dan Diajukan Untuk Memenuhi Salah Satu Syarat Kelulusan Program Studi Diploma III Teknik Informatika dan Mencapai Gelar Ahli Madya (A.Md.)

**Oleh :**

**ANDY FERBIANTORO**

**NIM. 361755401010**

**PROGRAM STUDI DIPLOMA III**

**TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANYUWANGI**

**2019**

---*Halaman ini sengaja dikosongkan---*

# LEMBAR PENGESAHAN

**PROPOSAL PROYEK AKHIR**

Judul : Pengembangan Modul Keuangan Aplikasi Pembukuan Notaris dan PPAT Menggunakan Yii Framework

Oleh : Andy Ferbiantoro

NIM : 361755401028

**Telah diuji pada :**

Hari :

Tanggal :

Tempat :

**Mengetahui / Menyetujui :**

|  |  |
| --- | --- |
| Dosesn Penguji :  1.  NIK. | Dosen Pembimbing :  1. Vivien Arief Wardhany, S.T., M.T.  NIP.198404032019032012 |
| 2.  NIK. | 2. Devit Suwardiyanto,S.Si.,M.T.  NIK. 198311052015041001 |

---*Halaman ini sengaja dikosongkan---*

# DAFTAR ISI

Halaman

SAMPUL LUAR

[HALAMAN JUDUL i](#_Toc30010964)

[LEMBAR PENGESAHAN iii](#_Toc30010965)

[DAFTAR ISI v](#_Toc30010966)

[BAB 1 PENDAHULUAN 1](#_Toc30010967)

[1.1 Latar Belakang 1](#_Toc30010968)

[1.2 Tujuan 2](#_Toc30010969)

[1.3 Rumusan Masalah 2](#_Toc30010970)

[1.4 Manfaat 2](#_Toc30010971)

[1.5 batasan Masalah 3](#_Toc30010972)

[BAB 2 TINJAUAN PUSTAKA 5](#_Toc30010973)

[2.1 Dasar Tori Penunjang 5](#_Toc30010974)

[2.1.1 Notaris 5](#_Toc30010975)

[2.1.2 Website 5](#_Toc30010976)

[2.1.3 *HTML* 5](#_Toc30010977)

[2.1.4 *HyperText Preprocessor (PHP)* 6](#_Toc30010978)

[2.1.5 *Yii Framework* 6](#_Toc30010979)

[2.1.5 *Web Service* 7](#_Toc30010980)

[2.16 *JSON* 7](#_Toc30010981)

[2.1.7 *MySQL* 8](#_Toc30010982)

[2.1.8 *Entity Relationship Diagram (ERD)* 8](#_Toc30010983)

[2.1.9 *Use Case* 8](#_Toc30010984)

[2.2 Pengujian 9](#_Toc30010985)

[2.3 Penelitian Terkait 10](#_Toc30010986)

[2.3.1 Sistem Informasi Administrasi Pada Kantor Notaris dan PPAT Hendro Winata, SH. 10](#_Toc30010987)

[2.3.2 Sistem Informasi Pengarsipan Pada Kantor Notaris Efrina Nofiyanti Kayadu, SH., M.Kn Berbasis Web Dengan Metode Waterfall 10](#_Toc30010988)

[2.3.3 Sistem Informasi Pada Kantor Notaris dan PPAT Juhaidi, S.H. 10](#_Toc30010989)

[1. BAB 3 METODE PENELITIAN 12](#_Toc30010990)

[3.1 Waktu, Tempat, dan Jadwal Penelitian 12](#_Toc30010991)

[3.1.1 Waktu Penelitian 12](#_Toc30010992)

[3.1.2 Tempat Penelitian 12](#_Toc30010993)

[3.1.3Jadwal Penelitian 12](#_Toc30010994)

[3.2 Metode Pengembangan Sistem 12](#_Toc30010995)

[3.2.1 Perencanaan Sistem 13](#_Toc30010996)

[3.2.2 Pembangunan sistem 13](#_Toc30010997)

[3.2.3 Demonstrasi Sistem 13](#_Toc30010998)

[3.2.4 Pengujian 13](#_Toc30010999)

[3.2.5 Implementasi 13](#_Toc30011000)

[3.2.6 Evaluasi 13](#_Toc30011001)

[3.3 Gambaran Umum Sistem 13](#_Toc30011002)

[3.3.1 Gambaran Umum Sistem yang Berjalan 13](#_Toc30011003)

[3.3.2 Gambaran Umum Sistem yang Diusulkan 13](#_Toc30011004)

[3.4 Desain Sistem 13](#_Toc30011005)

[3.4.1 *Use Case Diagram* 13](#_Toc30011006)

[3.5 Spesifikasi Sistem 14](#_Toc30011007)

[3.5.1 Struktur Menu 14](#_Toc30011008)

[3.5.2 Perencanaan Basis Data 14](#_Toc30011009)

[3.6 Desain Tampilan 14](#_Toc30011010)

---*Halaman ini sengaja dikosongkan---*

# BAB 1 PENDAHULUAN

## Latar Belakang

Notaris merupakan Pejabat umum yang bertugas membuat mengenai segala perbuatan, perjanjian atau penetapan sesuai peraturan yang ditetapkan oleh perundang-undangan. Berbeda dengan Pejabat Pembuat Akta Tanah, PPAT adalah seorang pejabat umum yang memiliki wewenang untuk membuat akta autentik tentang perbuatan tertentu mengenai hak atas tanah atau hak milik atas satuan rumah. Meski keduanya merupakan profesi yang berbeda, namun memungkinkan bagi seorang Notaris merangkap jabatan sebagai PPAT juga sebagaimana yang diatur dalam Pasal 17 huruf g UU No. 30 Tahun 2004 tentang Jabatan Notaris.

Selain melakukan pembuatan akta, seorang Notaris juga harus dapat melakukan manajemen keuangan yang tepat baik dari segi pemasukan maupun pengeluaran. Manajemen keuangan dilakukan guna mendapatkan informasi mengenai status bahwa kantor Notaris tersebut dapat dikatakan laba atau rugi. Manajemen keuangan yang dapat dilakukan sorang Notaris diantaranya membuat laporan mengenai *Gross Profit*, list piutang, list biaya operasional, serta manajemen kas kantor.

Manajemen keuangan pada kantor notaris kebanyakan masih dilakukan secara konvensional, baik dari segi pencatatan maupun penghitungan. Hal tersebut dirasa kurang efektif karena dapat mengakitbatkan kesalahan yang mungkin dapat terjadi yang dimana hal tersebut dapat menyulitkan manajemen pemasukan maupun pengeluaran di kantor Notaris. Masalah lain yang dihadapi seorang Notaris dalam manajemen keuangan yaitu masih bercampurnya antara keuangan kantor dengan keuangan Notaris itu sendiri, yang dimana hal ini dapat menyulitkan seorang Notaris untuk mengetahui perkembangan dari perusahaannya.

Secara umum mayoritas setiap kantor Notaris dipastikan memiliki akses internet untuk dapat melakukan komunikasi atau hal lain semacamnyaserta peralatan komputer guna menunjang kebutuhan kantor. Ditambah lagi staff dan klien kantor Notaris yang menggunakan komputer maupun perangkat mobile seperti *smartphone* untuk dapat terhubung ke internet. Menurut Data Statista 2019 menunjukkan pengguna internet di Indonesia pada 2018 sebanyak 95,2 juta, tumbuh 13,3% dari 2017 yang sebanyak 84 juta pengguna. Pada tahun selanjutnya pengguna internet di Indonesia akan semakin meningkat dengan rata-rata pertumbuhan sebesar 10,2% pada periode 2018-2023. Pada 2019 jumlah pengguna internet di Indonesia diproyeksikan tumbuh 12,6% dibandingkan 2018, yaitu menjadi 107,2 juta pengguna (Statista, 2019).

Berdasarkan permasalahan yang telah diuraikan serta potensi yang telah dijelaskan, maka diperlukan sebuah aplikas pembukuan akta yang dapat membantu seorang Notaris dalam melakukan manajemen keuangan baik dari segi pemasukan maupun pengeluaran. Dengan begitu Notaris dapat memantau untung dan rugi dari kantor serta perkembangan dari kantor Notaris itu sendiri.

## Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan diatas, didapatkan rumusan masalah sebagai berikut:

1. Bagaimana mengintegrasikan pembukuan akta dengan laporan keuangan?
2. Bagaimana mengimplementasikan metode manajemen akuntansi yang tepat kedalam sebuah aplikasi pembukuan akta?

## Tujuan

Berdasarkan identifikasi permasalahan yang telah dijelaskan pada latar belakang di atas, tujuan yang ingin dicapai adalah modul keuangan aplikasi pembukuan akta berbasis web yang berguna untuk pekerjaan Notaris dan PPAT khususnya dalam hal kegiatan manajemen pemasukan maupun pengeluaran keuangan serta laba dan rugi pada kantor Notaris.

## Manfaat

Manfaat dari perancangan proyek akhir ini yaitu mempermudah seorang Notaris dalam hal memanajemen keuangan baik dari segi pamasukan maupun pengeluaran sehingga sirkulasi keuangan pada kantor notaris menjadi lebih terkendali.

## batasan Masalah

Agar penyusunan Proyek Akhir ini lebih jelas dan tidak keluar dari pokok permasalahan, ada beberapa hal yang dijadikan sebagai batasan masalah antara lain:

1. Modul keuangan aplikasi pembukuan akta Notaris ini menangani manajemen keuangan kantor dari segi pemasukan maupun pengeluaran serta laba dan rugi.
2. Aplikasi ini dapat digunakan oleh Notaris, staff, dan client.
3. Bahasa pemrograman yang digunakan untuk membuat modul keuangan aplikasi pembukuan akta Notaris ini menggunakan Bahasa pemrograman *PHP* dan *Yii* sebagai framework serta *MySQL* sebagai *database*.

---*Halaman ini sengaja dikosongkan---*

# BAB 2 TINJAUAN PUSTAKA

## 2.1 Dasar Tori Penunjang

Guna mendukung pembuatan Proyek Akhir ini, maka perlu adanya landasan teori sebagai pengetahuan bagi penulis agar hasil pembuatan Proyek Akhir sesuai dengan yang diharapkan.

### 2.1.1 Notaris

Notaris merupakan Pejabat umum yang bertugas membuat mengenai segala perbuatan, perjanjian atau penetapan sesuai peraturan yang ditetapkan oleh perundang-undangan. Berbeda dengan Pejabat Pembuat Akta Tanah, PPAT adalah seorang pejabat umum yang memiliki wewenang untuk membuat akta autentik tentang perbuatan tertentu mengenai hak atas tanah atau hak milik atas satuan rumah. Meski keduanya merupakan profesi yang berbeda, namun memungkinkan bagi seorang Notaris merangkap jabatan sebagai PPAT juga sebagaimana yang diatur dalam Pasal 17 huruf g UU No. 30 Tahun 2004 tentang Jabatan Notaris.

### 2.1.2 Website

suatu halaman web yang saling berhubungan yang umumnya berisikan kumpulan informasi berupa data teks, gambar, animasi, audio, video maupun gabungan dari semuanya yang biasanya dibuat untuk personal, organisasi dan perusahaan. Dari pengertian website tersebut dapat dibedakan menjadi 2 yaitu web bersifat statis dan dinamis. Bersifat statis apabila isi informasinya tetap dan isi informasinya hanya dari pemilik website sedangkan web yang bersifat dinamis apabila isi informasinya selalu berubah-ubah dan dapat diubah-ubah oleh pemilik maupun pengguna website.

### 2.1.3 *HTML*

HTML adalah kepanjangan dari **H*ypertext Markup Language*** dan merupakan salah satu bahasa yang paling banyak digunakan dalam mebuat halaman web. Seperti namanya, HTML adalah bahasa Markup yang berarti HTML hanya digunakan untuk “mark-up” dokumen teks dengan tag yang akan memberitahukan browser struktur untuk menampilkan sebuah desain layout web. File HTML berupa teks sederhana yang memungkinkan kita dapat menggunakan teks editor sederhana untuk dapat menulisnya.

### 2.1.4 *HyperText Preprocessor (PHP)*

*HyperText Preprocessor* (PHP) merupakan bahasa pemrograman web *server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada server (*server side HTML embedded scripting*). PHP merupakan *script* yang umumnya digunakan dalam hal pembuatan halaman *website* yang dinamis, yang berarti halaman yang ditampilkan sesuai permintaan klien. Mekanisme seperti ini menyebabkan informasi yang didapat oleh klien selalu terbaru. PHP memliki fungsi *built-in* lengkap, cepat, mudah dipelajari, dan bersifat gratis. PHP dapat berjalan di berbagai *web server* dan sistem operasi yang berbeda. Selain itu integrasi antara PHP dengan basis data sangat mudah diimplementasikan, sebab PHP mendukung banyak DBMS (*Database Management System*) serta tidak memerlukan kompilasi dalam sumbernya. (Anhar, 2010)

### 2.1.5 *Yii Framework*

Yii Framework merupakan sebuah software yang digunakan sebagai kerangkakerja *PHP* yang berbasis komponen dengan performasi tinggi untuk mempermudah dalam membangun sebuah website berskala besar. Yii Framework menyediakan resuabilitasmaksimum dalam pemrograman web dan bisa mengakselerasi proses pengembangan secara signifikan.

Yii Framework juga merupakan MVC framework seperti kebanyakan framework lain. Yii telah memplementasikan pola desain MVC ( Model-View-Controller) yang diadopsi secara luas dalam pemrograman web. Selain implementasi MVC, Yii juga memperkenalkan front controller(controller depan), yang disebut Aplikasi, yang akan mengenkapsulasi konteks eksekusi untuk memproses sebuah request.

### 2.1.5 *Web Service*

*Web service* adalah aplikasi sekumpulan data (*database*), perangkat lunak (*software*) atau bagian dari perangkat lunak yang dapat diakses secara remote oleh berbagai piranti dengan sebuah perantara tertentu.  Secara umum, *web service*  dapat diidentifikasikan dengan menggunakan URL seperti hanya web pada umumnya. *Web service* dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan dimana sebuah *database* ditanamkan, dibuat dalam bahasa apa sebuah aplikasi yang mengkonsumsi data, dan di platform apa sebuah data itu dikonsumsi. *Web service* mampu menunjang interoperabilitas. Sehingga *web service* mampu menjadi sebuah jembatan penghubung antara berbagai sistem yang ada. Namun yang membedakan *web service* dengan web pada umumnya adalah interaksi yang diberikan

oleh *web service*. Berbeda dengan URL *web* pada umumnya, URL *web service* hanya menggandung kumpulan informasi, perintah, konfigurasi atau sintaks yang berguna membangun sebuah fungsi-fungsi tertentu dari aplikasi ( Anindita Saktiaji, 2015).

### 2.16 *JSON*

*JSON* merupakan singkatan dari ***JavaScript Object Notation*** merupakan format untuk pertukaran data yang diturunkan dari Bahasa pemrograman *javaScript*, akan tetapi format ini tersedia bagi banyak bahasa lain termasuk *Python, Ruby, PHP*, dan *Java*. *JSON* mengguanakan ekstensi *.json* saat dia berdiri sendiri Saat didefinisikan di dalam format file lain (seperti di dalam .html), dia dapat tampil didalam tanda petik sebagai *JSON* string, atau ia dapat dimasukkan kedalam sebuah variabel.

Dalam sebuah objek *JSON* adalah format data *key – value* yang biasanya diproses didalam kurung kurawal ({}). ***Key****JSON* berada di sebelah kiri tanda titik dua. Mereka perlu dibungkus oleh tanda petik dua dan dapat berupa string apapun yang valid. **Value** *JSON* ada di sebelah kanan tanda titik dua. Ada enam tipe data dasar yang bisa dipakai untuk mengisinya diantaranya *String, Numbers, objects, arrays, Boolean (true atau false), null.*

### 2.1.7 *MySQL*

MySQL merupakan perangkat lunak sistem amnajemen basis data SQL atau sering disebut sebagai *Database Management System (DBMS).* MySQL merupakan implementasi dari system manajemen basisdata relasional (RDBMS). Saat ini MySQL merupakan salah satu database server yang sangat terkenal di dunia, MySQL menggunakan yang digunakan untuk mengakses database yaitu SQL. SQL (*Structured Query Language*) pertama kali diterapkan pada sebuah proyek riset pada laboratorium riset San Jose, IBM yang bernama system R. Kemudian SQL juga dikembangan oleh Oracle, Informix dan Sybase.SQL banyak digunakan karena proses pengaksesan database lebih user-friendly.

### 2.1.8 *Entity Relationship Diagram (ERD)*

*Entity Relationship Diagram (ERD)* merupakan salah satu permodelan ynag digunakan untuk mendesain sebuah database denga tujuan untuk menjelaskan atau menggambarkan relasi dalam database. Notasi – notasi simbolik dalam ERD yang dapt kita gunakan dijelaskan pada Tabel 2.1.

**Tabel 2.0.1** Simbol – symbol dalam ERD

|  |  |  |  |
| --- | --- | --- | --- |
| No | Simbol | Nama | Keterangan |
| 1 |  | *Entity* | Merupakan objek yang dapat dibedakan dengan objek lainnya yang berfungsi memberikan identitas pada entitas yang memiliki label atau nama. |
| 2 |  | Atribut | Berfungsi untuk memperjelas atribut yang dimiliki oleh sebuah entitas |
| 3 |  | Relasi | Berfungsi untuk mengetahui jenis hubungan yang ada antara 2 entitas |
| 4 |  | Alur | Berfungsi untuk menghubungkan atribut dengan entitas dan entitas dengan relasi |

### 2.1.9 *Use Case*

*Use Case* merupakan sebuah teknik yang digunakan dalam pengembangan sebuah software atau sistem informasi untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan, Use Case menjelaskan interaksi yang terjadi antara ‘aktor’ — inisiator dari interaksi sistem itu sendiri dengan sistem yang ada, sebuah Use Case direpresentasikan dengan urutan langkah yang sederhana (Arifwicaksanaa, 2016). Adapun symbol yang digunakan pada *use case* diagram dijelaskan pada Tabel 2.2

**Tabel 2.0.2** Simbol –simbol Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| No | Simbol | Nama | Keterangan |
| 1 |  | Aktor | Aktor adalah orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat,. |
| 2 |  | *Use Case* | Use case adalah fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor. |
| 3 |  | *Include* | Menunujukkan bahwa *use case* merupakan fungsionalitas dari *use case* yang lain |
| 4 |  | *Extend* | Menunjukkan bahwa *use case* merupakan tambahan fungsional dari *use case* lain jika suatu kondisi terpenuhi |
| 5 |  | *Association* | Menunjukkan komunikasi antara aktor dan use case yang berpartisipasi pada use case diagram atau use case yang memiliki interaksi dengan aktor. |
| 6 |  | *Generalization* | Elemen yang menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan *use case* |
| 7 |  | *System* | Menspesifikasikan paket yang menampilkan sistem secara terbatas |

## 2.2 Pengujian

*White Box Test*ing Merupakan pengujian yang didasarka pada detail prosedur dan alur detail code program, pada pengujian White Box dilakukank test case untuk mencari kesalahan / bugs secara detail dari kode program yang dibuat.

*Black Box Testing* Merupakan pengujian yang didasarkan pada fungsionalitas dari sistem yang dibuat seperti fitur – fitur, kesesuaian alur dan fungsi dengan proses yang diharapkan. Dengan kata lain pengujian ini memfokuskan pada fungsionalitas dari sistem yang dibuat apakah sudah sesuai atau masih memrlukan perbaikan.

## 2.3 Penelitian Terkait

Untuk mendukung proses perancangan Proyek Akhir ini, maka diperlukan penelitian terkait yang digunakan untuk perbandingan pembuatan proyek akhir ini agar hasil pembuatan Proyek Akhir sesuai dengan yang diharapkan.

### 2.3.1 Sistem Informasi Administrasi Pada Kantor Notaris dan PPAT Hendro Winata, SH.

Penelitian ini disusun oleh Fikri Akbar Herdianto dan Meme Sulilowati, S.Kom.,MMSI. pada tahun 2018. Penelitian ini bertujuan untuk membangun sebuah sistem informasi berbasis web untuk memudahkan klien dalam memantau proses pembuatan akta dari perkiraan biaya , hingga notifikasi yang diberikan saat akta telah selesai dan juga membantu kantor dalam melakukan pembukuan dan menjaga dokumentasi akta yang pernah dibuat serta pendataan klien yang pernah membuat akta di Kantor Notaris dan PPAT Hendro Winata, SH.

### 2.3.2 Sistem Informasi Pengarsipan Pada Kantor Notaris Efrina Nofiyanti Kayadu, SH., M.Kn Berbasis Web Dengan Metode Waterfall

Peneitian ini disusun oleh Romindo dan Novia Ameylia pada tahun 2019. Hasil dari penelitia ini adalah membangun sebuah sistem informasi yang memudahkan pegawai maupun notaris di kantor Notaris Efrina Nofiyanti Kayadu, SH., M.Kn dalam menyimpan dan mencari arsip berkas dari klien.

### 2.3.3 Sistem Informasi Pada Kantor Notaris dan PPAT Juhaidi, S.H.

Penelitian ini dirancang oleh Kristanto, Yohanes Fernando, dan Desy Iba Ricoida pada tahun 2018. Tujuan dari penelitian ini adalah membangun sebuah sistem informasi berbasis web yang berfungsi untuk memudahkan kantor dalam mengelola dokumen dan membuat pencarian data klien menjadi kebih cepat dan efisien serta memudahkan pemberian informasi prosedur pembuatan surat kepada pemohon. Pemohon dapat menginput sendiri data yang diperlukan untuk membuat surat tersebut secara online.

---*Halaman ini sengaja dikosongkan---*

# BAB 3 METODE PENELITIAN

**METODE PENELITIAN**

## 3.1 Waktu, Tempat, dan Jadwal Penelitian

### 3.1.1 Waktu Penelitian

### 3.1.2 Tempat Penelitian

### 3.1.3Jadwal Penelitian

Adapun jadwal kegiatan proyek akhir pada tahun ajaran 2019/2020 ditunjukkan pada Tabel 3.1.

Tabel 3.1 Jadwal Penelitian

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## 3.2 Metode Pengembangan Sistem

Metode yang digunakan dalam pengerjaan proyek akhir ini menggunakan metode *Rapid Application Development* (RAD). Metode RAD merupakan proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan dalam waktu singkat. Selain itu RAD menggunakan metode berulang dalam mengembangkan sistem dimana model kerja sistem dikontruksikan pada tahap awal pembangunan sistem dengan tujuan menetapkan kebutuhan pengguna secara tepat. RAD menekankan pada siklus pembangunan. pendek, singkat, dan cepat. Jadi, metode RAD sesuai untuk menghasilkan sistem perangkat lunak dengan kebutuhan yang mendesak dan waktu yang singkat dalam penyelesaiannya. Metode RAD yang digunakan terdapat Pada Gambar 3.1.



Gambar 3.1 Metode Rapid Aplication Development

### 3.2.1 Perencanaan Sistem

### 3.2.2 Pembangunan sistem

### 3.2.3 Demonstrasi Sistem

### 3.2.4 Pengujian

### 3.2.5 Implementasi

### 3.2.6 Evaluasi

## 3.3 Gambaran Umum Sistem

### 3.3.1 Gambaran Umum Sistem yang Berjalan

### 3.3.2 Gambaran Umum Sistem yang Diusulkan

## 3.4 Desain Sistem

### 3.4.1 *Use Case Diagram*

## 3.5 Spesifikasi Sistem

### 3.5.1 Struktur Menu

### 3.5.2 Perencanaan Basis Data

## 3.6 Desain Tampilan